

Trabalho Prático 3 – Índice baseado em Árvores de Pesquisa

Valor: 10 pontos

Data de devolução: 05/12/2006

Pode ser feito em duplas.

Faça um programa que leia um texto qualquer (arquivo no formato *.txt*) e imprima, em ordem alfabética, todas palavras com 3 ou mais caracteres e a linha na qual elas aparecem no texto. Por exemplo, para o texto:

```
Um, dois, tres, testando.  
Testando tres vezes.  
Um teste final.
```

A saída deve ser algo como:

```
dois      1  
final     3  
testando  1,2  
teste     3  
tres      1,2  
vezes     2
```

A leitura do arquivo deverá desprezar espaços em branco e sinais de pontuação, que serão considerados separadores de palavras. Além disso, a leitura deverá converter todas as letras maiúsculas em minúsculas. Você pode considerar que cada palavra contém no máximo 20 letras e que não haverá caracteres acentuados.

Para isso, você deverá implementar uma *Árvore de Pesquisa sem Balanceamento*, onde cada nodo irá armazenar uma palavra e uma lista de números das linhas onde ela aparece (use o TAD Lista para isso). A medida que as palavras são lidas, o seu algoritmo deve pesquisar a árvore para ver se a palavra já está presente. Se estiver, adiciona o novo número de linha à lista dessa palavra. Se não estiver presente, cria um novo nodo na árvore e inicia a lista de linhas com esta. O uso da árvore de pesquisa é indicado pois a sua estrutura será bastante dinâmica e no final você precisará imprimir as palavras em ordem alfabética.

Para executar seu programa, use parâmetros de linha de comando para fazer sua chamada. Esse tipo de execução é bastante comum em sistemas Unix / Linux e no antigo DOS. Por exemplo, se o seu programa chama-se **prog** e você quiser montar o índice do arquivo texto.txt, sua chamada deve ser:

```
> prog texto.txt
```

Descubra em sua linguagem de preferência como ler os parâmetros da linha de comando. Também faz parte do trabalho descobrir como fazer a leitura de um arquivo texto e a manipulação adequada de strings em sua linguagem de programação de preferência.

O que deve ser entregue:

- Código fonte do programa em Pascal ou C (bem identada e comentada).
- Impressão dos resultados obtidos com os dois arquivos de teste disponibilizados em:
<http://www2.dcc.ufmg.br/disciplinas/aeds2/faroeste.txt>
<http://www2.dcc.ufmg.br/disciplinas/aeds2/brasil.txt>
- Programa executável compilado em um dos seguintes compiladores: Turbo Pascal, Turbo C, Dev Pascal ou Dev C. Indique na documentação o compilador usado.
- Documentação do trabalho. Entre outras coisas, a documentação deve conter:
 1. Introdução: descrição do problema a ser resolvido e visão geral sobre o funcionamento do programa.
 2. Implementação: descrição sobre a implementação do programa. Deve ser detalhada a estrutura de dados utilizada (de preferência com diagramas ilustrativos), o funcionamento das principais funções e procedimentos utilizados, o formato de entrada e saída de dados, bem como decisões tomadas relativas aos casos e detalhes de especificação que porventura estejam omissos no enunciado.
 3. Estudo de Complexidade: estudo da complexidade do tempo de execução dos procedimentos implementados e do programa como um todo (notação O), considerando a matriz de tamanho n .
 4. Conclusão: comentários gerais sobre o trabalho e as principais dificuldades encontradas em sua implementação.
 5. Bibliografia: bibliografia utilizada para o desenvolvimento do trabalho, incluindo sites da Internet se for o caso

Obs2: Consulte as dicas do Prof. Nívio Ziviani de como deve ser feita uma boa implementação e documentação de um trabalho prático: <http://www.dcc.ufmg.br/~nivio/cursos/aed2/roteiro/>

Comentários Gerais:

- 1 Comece a fazer este trabalho logo, enquanto o problema está fresco na memória e o prazo para terminá-lo está tão longe quanto jamais poderá estar.
- 2 Clareza, indentação e comentários no programa também vão valer pontos.
- 3 O trabalho pode ser feito em duplas
- 4 Trabalhos copiados serão penalizados conforme anunciado
- 5 Penalização por atraso: $(2^d - 1)$ pontos, onde d é o número de dias (úteis) de atraso. Note que após dois dias úteis, o trabalho não pode mais ser entregue.